Robust Parallel Algorithm for Anisotropic Adaptive Tetrahedral Meshes

Konstantin Lipnikov **Yuri Vassilevski**

LANL
Los Alamos, NM
lipnikov@lanl.gov

INM/LAM1
Moscow/Augsburg
vasilevs@dodo.inm.ras.ru

Contents

- Objectives
- Quasi-optimal meshes
- Parallel mesh adaptation
- Numerical experiments
- Conclusions

Objectives

Notation and assumptions.

- $u \in C^2(\Omega)$ (may be relaxed to $u \in W_1^2(\Omega)$);
- Ω^h is a conformal mesh consisting of simplexes;
- N_T is a fixed number of simplexes.

Definition. Ω^h is the optimal mesh if

$$\Omega^h = \arg \min_{\Omega^h : \#T = N_T} \|u - P^h u\|_{L_\infty(\Omega)}.$$

Existence. If $\|u - P^h u\|_{L_\infty(\Omega)}$ is (a) continuous functional of the nodes coordinates and (b) non-increasing functional for the case of nested grids, then the optimal triangulation consisting of N_T simplexes *exists*.

- $u \in C^2(\bar{\Omega})$ and P^h is the piece-wise linear interpolation operator, i.e. $P^h = P_{\Omega^h}$.

Quasi-optimal meshes (1/4)

Let G be a constant metric defined on a simplex Δ and $h^\star$ be a real positive number.

- Quality of a triangle Δ in metric G is defined by

$$Q_{G,h^\star}(\Delta) = 12\sqrt{3}\,\frac{|\Delta|_G}{|\partial\Delta|_G^2}\,F\left(\frac{|\partial\Delta|_G}{3h^\star}\right)$$

where

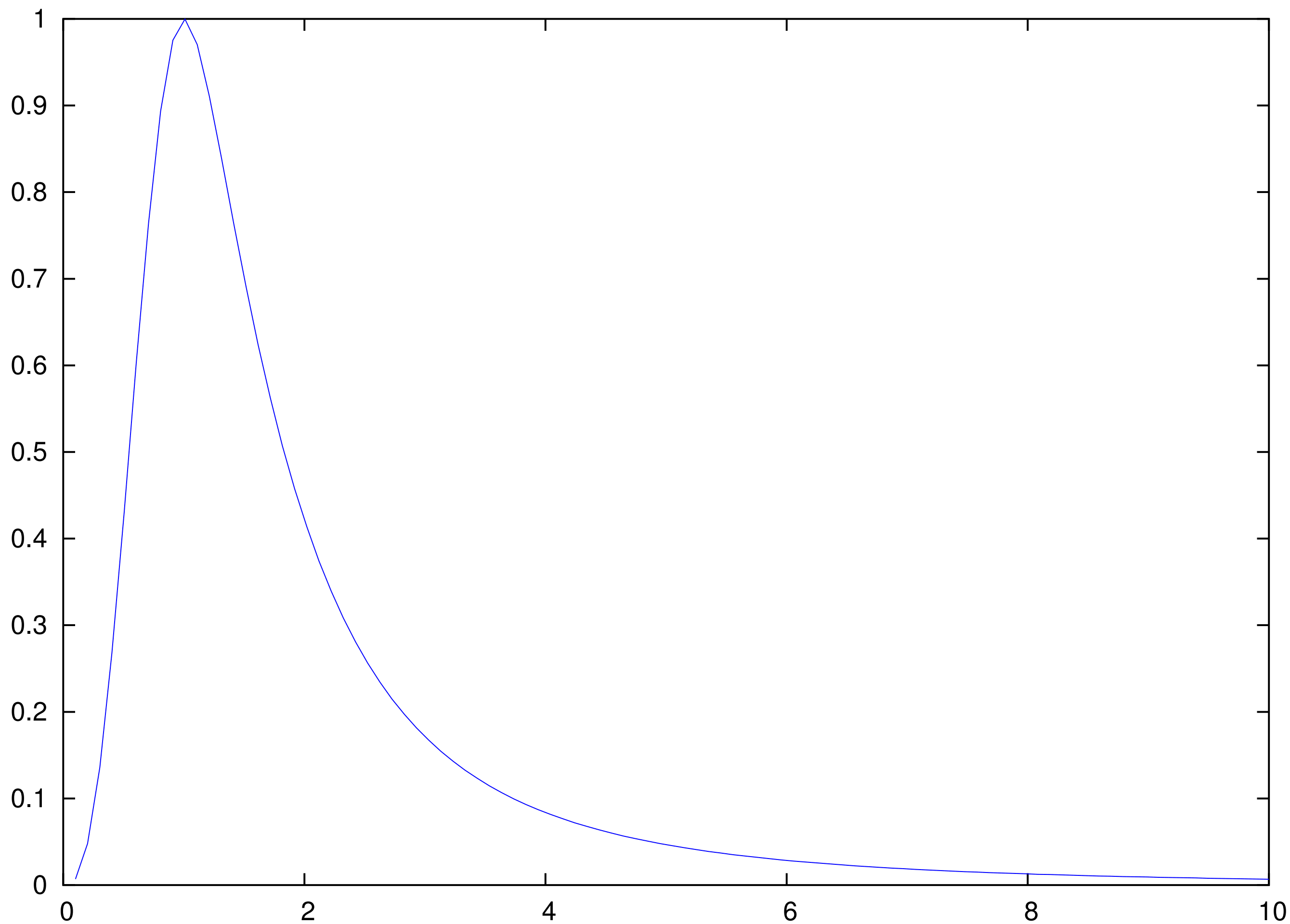
$$F(x) = \left(\min\left\{x, \frac{1}{x}\right\}\left(2 - \min\left\{x, \frac{1}{x}\right\}\right)\right)^3.$$

- Quality of a tetrahedron Δ in metric G is defined by

$$Q_{G,h^\star}(\Delta) = 6^4\sqrt{2}\,\frac{|\Delta|_G}{|\partial\partial\Delta|_G^3}\,F\left(\frac{|\partial\partial\Delta|_G}{6h^\star}\right).$$

Quasi-optimal meshes (2/4)

Function $F(x)$.



Quasi-optimal meshes (3/4)

The quality of triangulation Ω^h consisting of N_T simplexes in metric $G(x)$ is

$$Q_{G,N_T}(\Omega^h) = \min_{\Delta \in \Omega^h} Q_{G_\Delta, h^\star}(\Delta)$$

where

$$G_\Delta = G(\arg \max_{x \in \Delta} detG(x))$$

and

$$h^\star = \sqrt{\frac{4|\Omega|_G}{\sqrt{3}N_T}} \qquad \left(h^\star = \sqrt[3]{\frac{12|\Omega|_G}{\sqrt{2}N_T}} \quad \text{in} \;\; 3D \right).$$

(P. Zavattieri, E. Dari, and G. Buscaglia,1996)

Definition. Triangulation Ω^h consisting of N_T simplexes is called quasi-optimal (with respect to u) if

$$Q_{|H|,N_T}(\Omega^h) \geq Q_0, \quad Q_0 = O(1),$$

where H is the Hessian (*matrix of second derivatives*) of u.

- Existence of a QOM depends on the value of Q_0.

Quasi-optimal meshes (4/4)

Quasi-optimal mesh Ω^h is an approximation to the optimal mesh Ω^h_{opt}.

Let $det H(x) \neq 0 \quad \forall x \in \Omega$ and

$$\|H_{ps} - H_{\Delta,ps}\|_{L_\infty(\Delta)} < q|\lambda_1(H_\Delta)|, \qquad 0 < q < 1/2,$$

for all $\Delta \in \Omega^h_{opt}$ and $\Delta \in \Omega^h$ where the maximal error is attained. Then

$$\|u - P_{\Omega^h} u\|_{L_\infty(\Omega)} \leq C(Q_0, q)\|u - P_{\Omega^h_{opt}} u\|_{L_\infty(\Omega)}.$$

Both the optimal mesh and quasi-optimal meshes satisfy:

$$C_1(Q_0, q)\frac{|\Omega|_{|H|}}{N_T} \leq \|u - P_{\Omega^h} u\|_{L_\infty(\Omega)} \leq C_2(Q_0, q)\frac{|\Omega|_{|H|}}{N_T} \qquad \text{(in } 2D),$$

$$C_1(Q_0, q)\left(\frac{|\Omega|_{|H|}}{N_T}\right)^{2/3} \leq \|u - P_{\Omega^h} u\|_{L_\infty(\Omega)} \leq C_2(Q_0, q)\left(\frac{|\Omega|_{|H|}}{N_T}\right)^{2/3} \qquad \text{(in } 3D)$$

Mesh adaptation algorithm

Initialization Step. Generate an initial triangulation Ω^h. Choose the final mesh quality Q_0, $Q_0 < 1$, and the final number N_T of mesh elements.

Iterative Step.

1. Compute the discrete solution $P^h u$ for triangulation Ω^h.
2. Recover the discrete Hessian H^h from $P^h u$.
   Stop iterations if $Q_{|H^h|,N_T}(\Omega^h) \geq Q_0$.
3. Generate the next mesh $\tilde{\Omega}^h$ such that $Q_{|H^h|,N_T}(\tilde{\Omega}^h) \geq Q_0$.
4. Set $\Omega^h = \tilde{\Omega}^h$ and go to 1.

- convergence analysis of the iterative step can be found in
  Comput. Math. Math. Phys., V.39, No.9, 1999, pp.1468–1468.
  East-West Journal, V.7, No.4, pp.223–244.

Mesh adaptation algorithm

Initialization Step. Generate an initial triangulation Ω^h. Choose the final mesh quality Q_0, $Q_0 < 1$, and the final number N_T of mesh elements.

Iterative Step.

1. Compute the discrete solution $P^h u$ for triangulation Ω^h.
2. Recover the discrete Hessian H^h from $P^h u$.
   Stop iterations if $Q_{|H^h|,N_T}(\Omega^h) \geq Q_0$.
3. Generate the next mesh $\tilde{\Omega}^h$ such that $Q_{|H^h|,N_T}(\tilde{\Omega}^h) \geq Q_0$.
4. Set $\Omega^h = \tilde{\Omega}^h$ and go to 1.

- convergence analysis of the iterative step can be found in
  Comput. Math. Math. Phys., V.39, No.9, 1999, pp.1468–1468.
  East-West Journal, V.7, No.4, pp.223–244.

Parallel mesh adaptation (1/4)

Assumptions.

1. Each processor may keep the global mesh.
2. Parallel computer has a few processors.
3. A mesh can be easily distributed among processors and gathered back.

mesh with 10^6 tets require about 34M of processor memory

Parallel mesh adaptation (2/4)

Algorithm of generation of $|H^h|$-QOM.

Initialization Step. Processor root computes and broadcasts the discrete Hessian H^h to other processors. Set $k = 1$. Processor root computes three orthogonal directions, $\vec{b}_k$, $k = 1, 2, 3$, of the inertia tensor of Ω^h.

Decomposition Step ($k < 4$). Processor root extracts mesh elements such that $Q_{|H^h|,h^\star}(\Delta) < Q_0$ and their neighbors. The extracted mesh is colored slice-wise ($\perp \vec{b}_k$) and broadcasted to other processors.

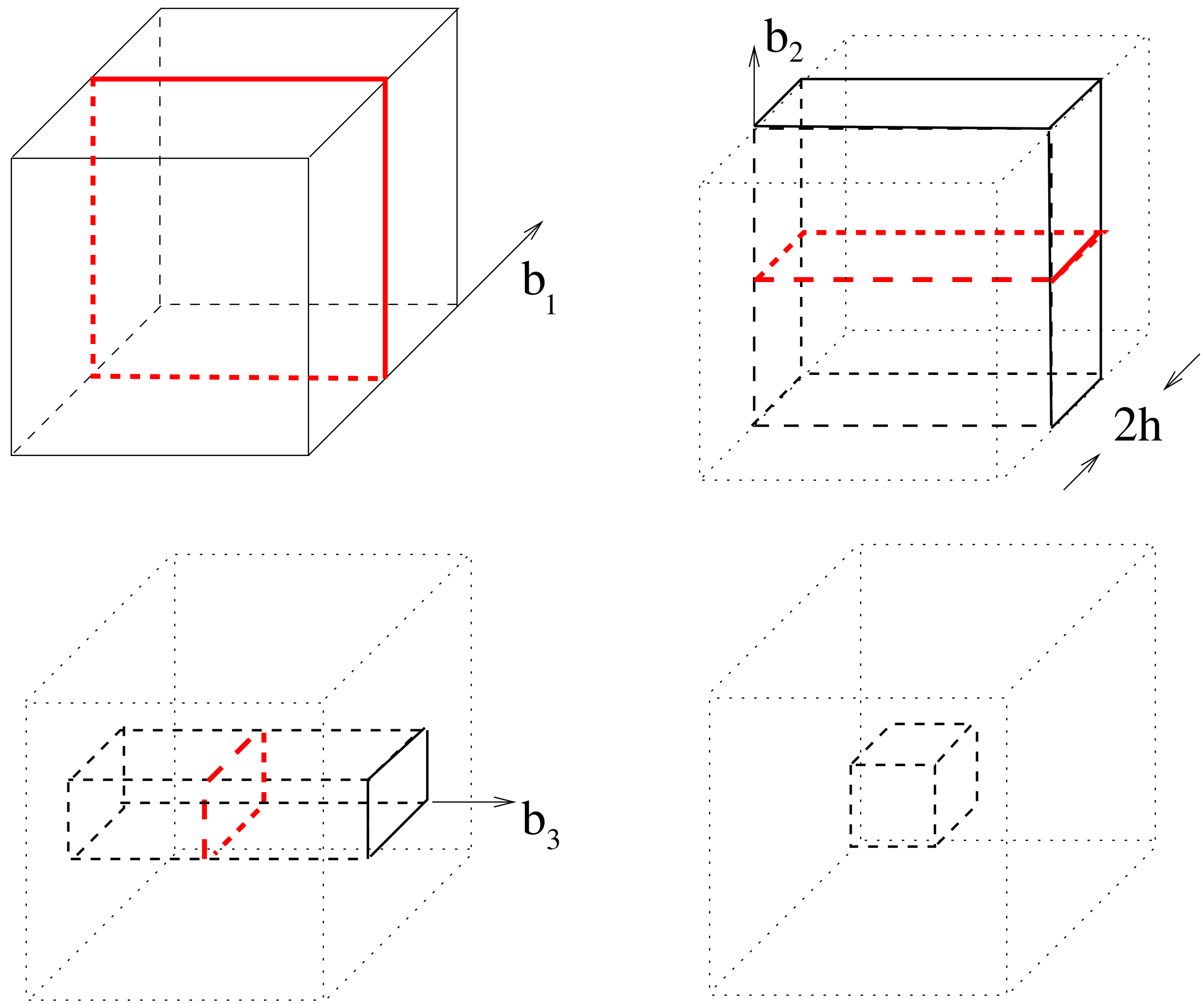
Decomposition Step ($k = 4$). Processor root extracts the mesh elements whose vertices were fixed for $k = 1, 2, 3$. All the extracted mesh elements are assigned to the processor root.

Generation Step. Processor p extracts the p-th subgrid and tries to construct a $|H^h|$-quasi-optimal mesh. The boundary triangles shared by any two subgrids are not modified.

Gathering Step. Processor root gathers the subgrids and builds a conforming global grid $\tilde{\Omega}^h$. We stop if $Q_{|H^h|,N_T}(\tilde{\Omega}^h) \geq Q_0$; otherwise, we set $k := k + 1$, $\Omega^h = \tilde{\Omega}^h$ and go to Decomposition Step.

Parallel mesh adaptation (3/4)

Decomposition steps for the case of 2 processors:



Parallel mesh adaptation (4/4)

To construct a $|H^h|$-quasi-optimal mesh we generate a sequence of grids

$$\Omega^h,\ \Omega_1^h,\ \Omega_2^h, \ldots, \Omega_{l_{\max}}^h$$

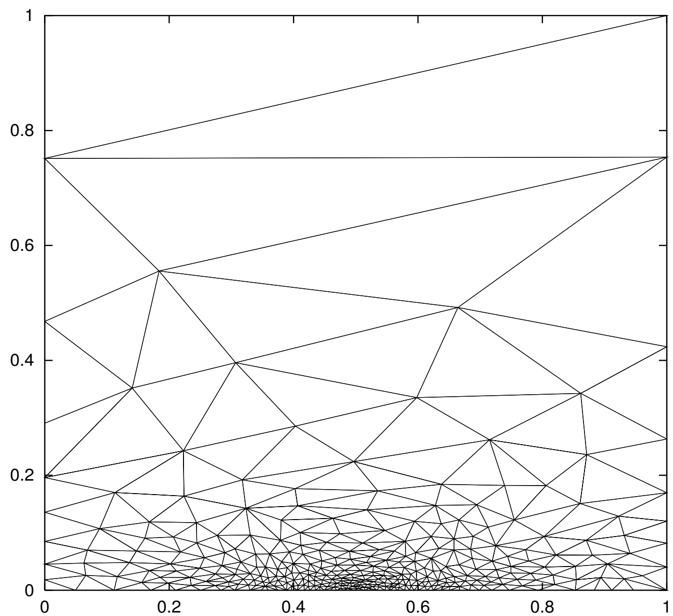
such that

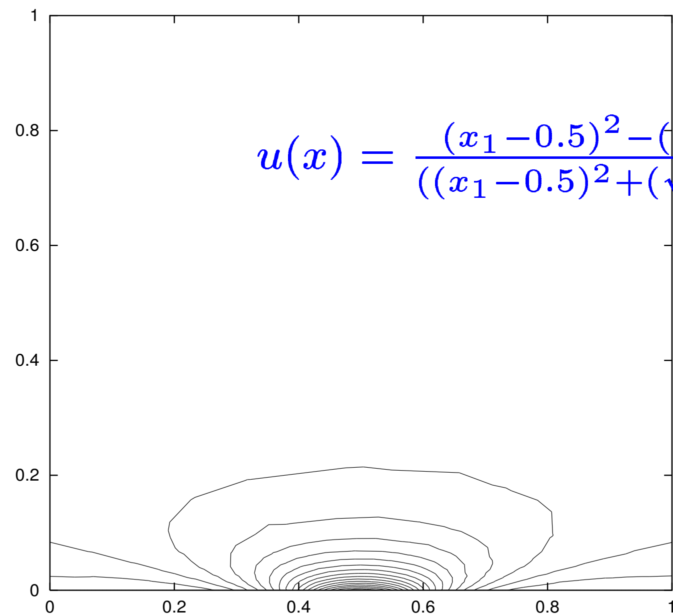
$$Q_{|H^h|,N_T}(\Omega^h) \leq Q_{|H^h|,N_T}(\Omega_1^h) \leq \ldots \leq Q_{|H^h|,N_T}(\Omega_{l_{\max}}^h).$$

- Take the worst simplex with its neighbors.
- Try to apply one of admissible mesh modifications (**add a point, swap face to edge, delete a point, move a point**) to increase $Q_{|H^h|,N_T}(\Omega_l^h)$.
- If all operations fail, we add the simplex to a list of failed simplexes. If the list is too big, all failed simplexes are released.

Numerical experiments (1/8)

$$10\frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2} = 0 \qquad \text{in} \quad \Omega = (0,1)^2.$$





$$u(x) = \frac{(x_1-0.5)^2-(\sqrt{10}x_2+0.2)^2}{((x_1-0.5)^2+(\sqrt{10}x_2+0.2)^2)^2}$$

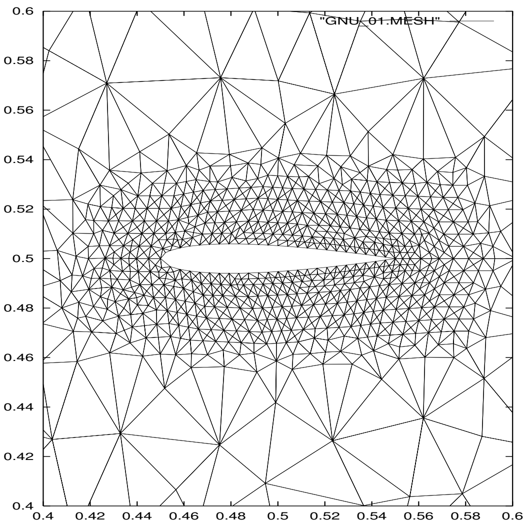
	Ω_7^h $Q_{\lvert H_k\rvert,N_T}(\Omega_k^h) \simeq 0.1$	$\hat{\Omega}^h$ optimal	$\tilde{\Omega}^h$ $Q_{\lvert H\rvert,N_T} = 1$	$\breve{\Omega}^h$ PLTMG nodes
$\#Tr$	600	608	569	686
$\varepsilon_{\max}$	0.216	0.065	0.167	0.404
ε_{mean}	0.067	0.037	0.063	0.086

Numerical experiments (2/8)

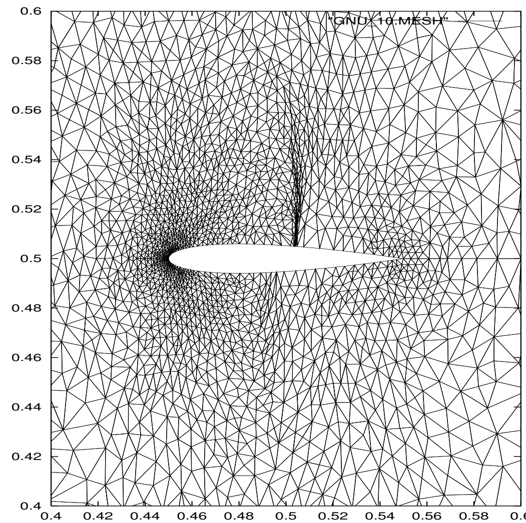
2D experiment: compressible irrotational isotropic adiabatic flow of an ideal gas around a wing.

$$
\begin{array}{rcll}
\operatorname{div}\left(1-\dfrac{|\nabla u|^2}{c}\right)^{\alpha}\nabla u & = & 0 & \text{in} \quad \Omega \\
\partial u/\partial n & = & \mathbf{v}_\infty\cdot\mathbf{n} & \text{on} \quad \Gamma_{ext} \\
\partial u/\partial n & = & 0 & \text{on} \quad \Gamma_{wing} \\
[\partial u/\partial \tau]=0, \quad [u] & = & Cir & \text{on} \quad \Gamma_{slit}
\end{array}
$$

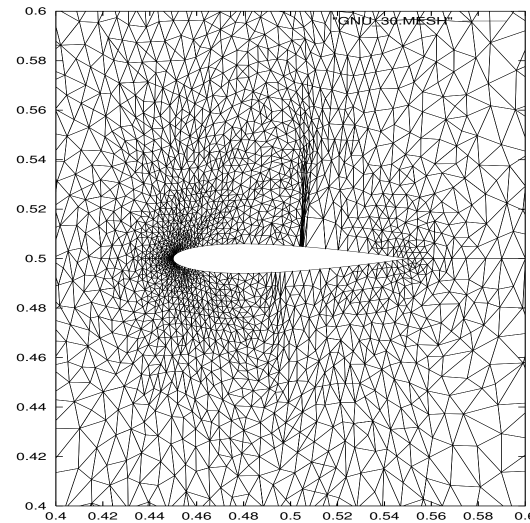
Numerical experiments (3/8)



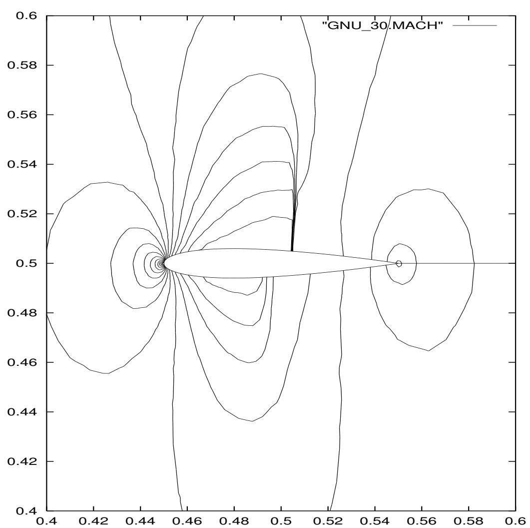
$itr\ 0, N_E = 3716$

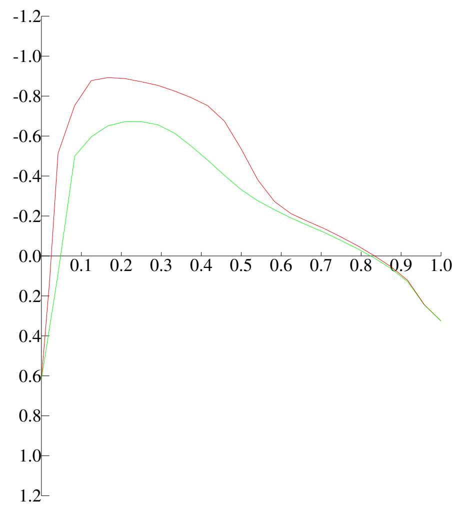


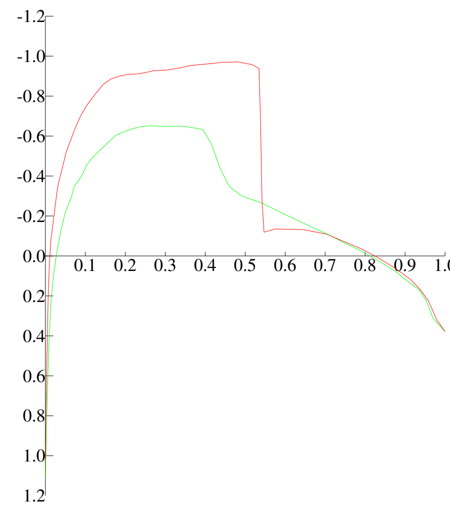
$itr\ 10, N_E = 5772$



$itr\ 30, N_E = 5754$







Numerical experiments (4/8)

3D experiment: point and anisotropic edge singularities.

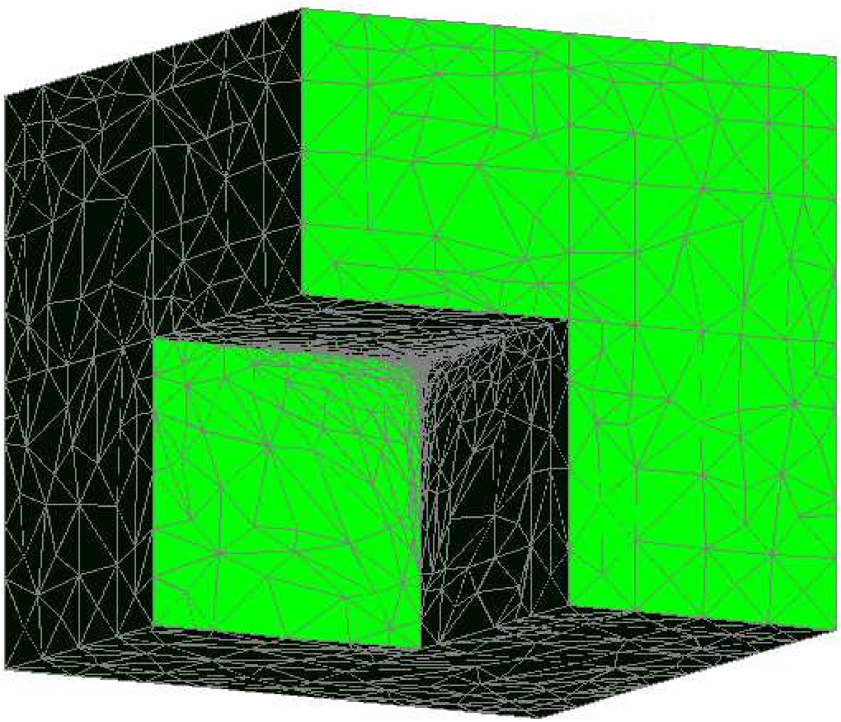
$$
\begin{aligned}
-\Delta u &= f \qquad \text{in} \quad \Omega = (0,1)^3 \setminus [0,0.5]^3 \\
u &= 0 \qquad \text{on} \quad \partial\Omega \\
f(x) &= \frac{1}{|x - x_0|}, \qquad x_0 = (0.5, 0.5, 0.5)
\end{aligned}
$$

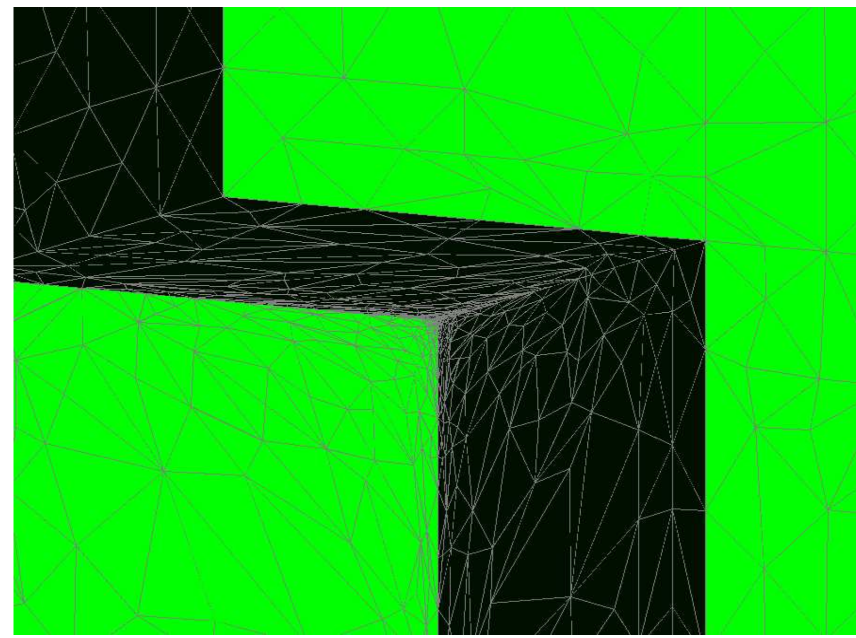
3D experiment: anisotropic boundary layers.

$$
\begin{aligned}
-10^{-2}\Delta u + \frac{\partial u}{\partial x_1} &= 1 \qquad \text{in} \quad \Omega = (0,1)^3 \setminus [0,0.5]^2 \times (0,1) \\
u &= 0 \qquad \text{on} \quad \partial\Omega
\end{aligned}
$$

Numerical experiments (5/8)

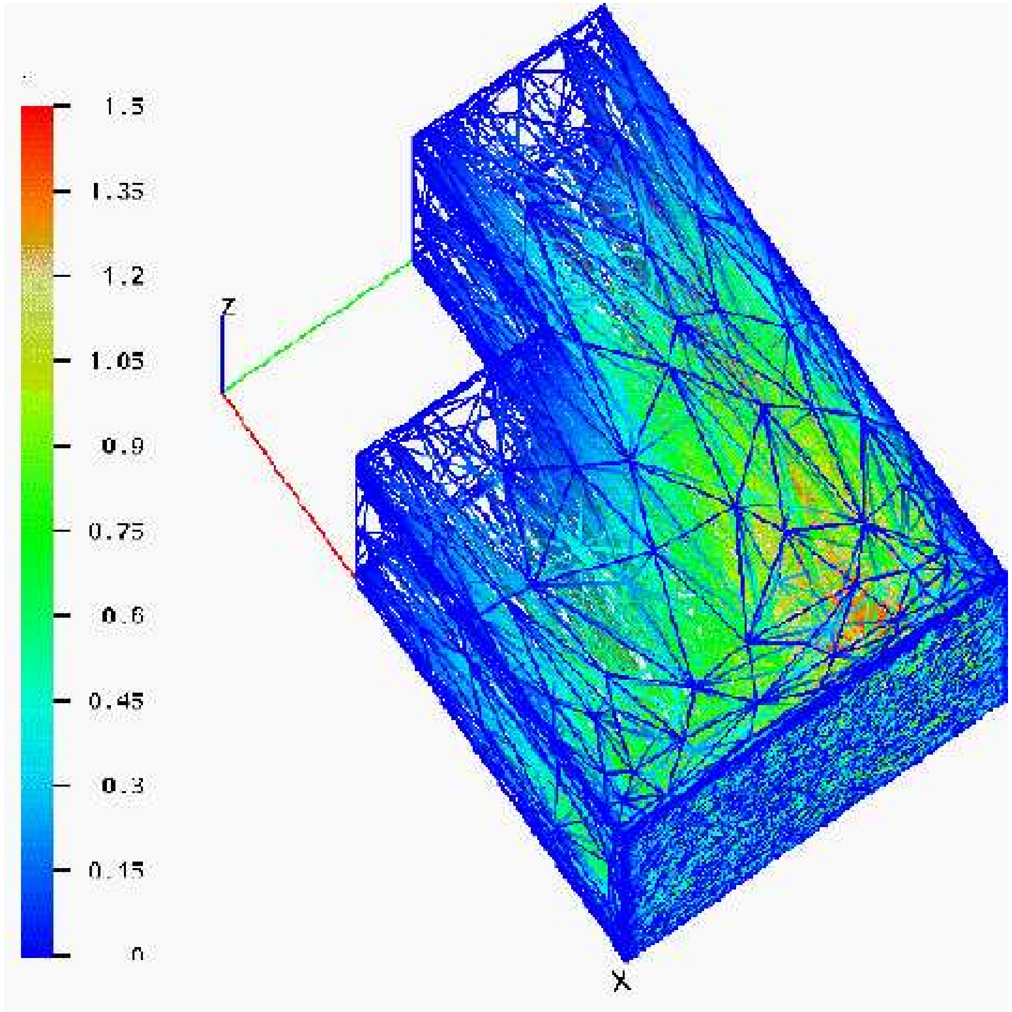
Adaptive grid for the 1st problem.





Numerical experiments (6/8)

Quasi-optimal mesh for the 2nd problem colored by solution values.



- Maximal aspect ratio of tetrahedra is about 100.

Numerical experiments (7/8)

Mesh refinement:

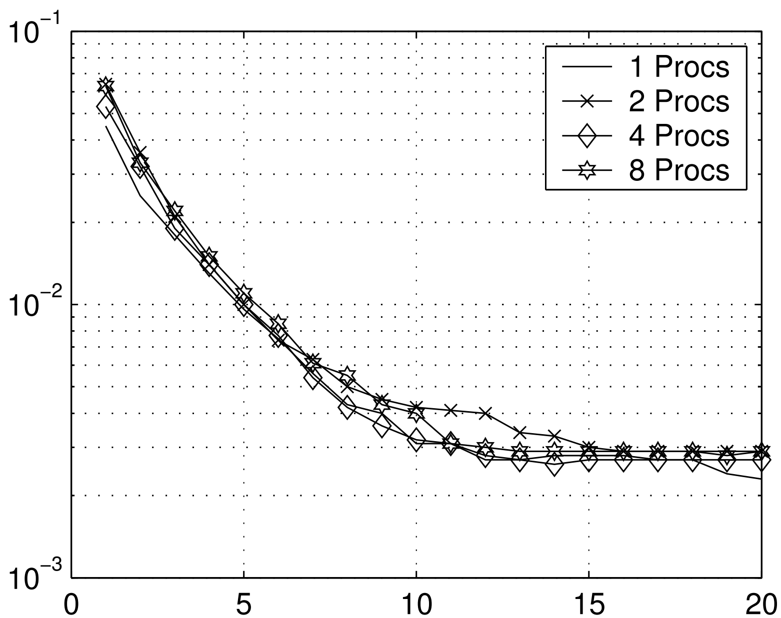
Problem 1:

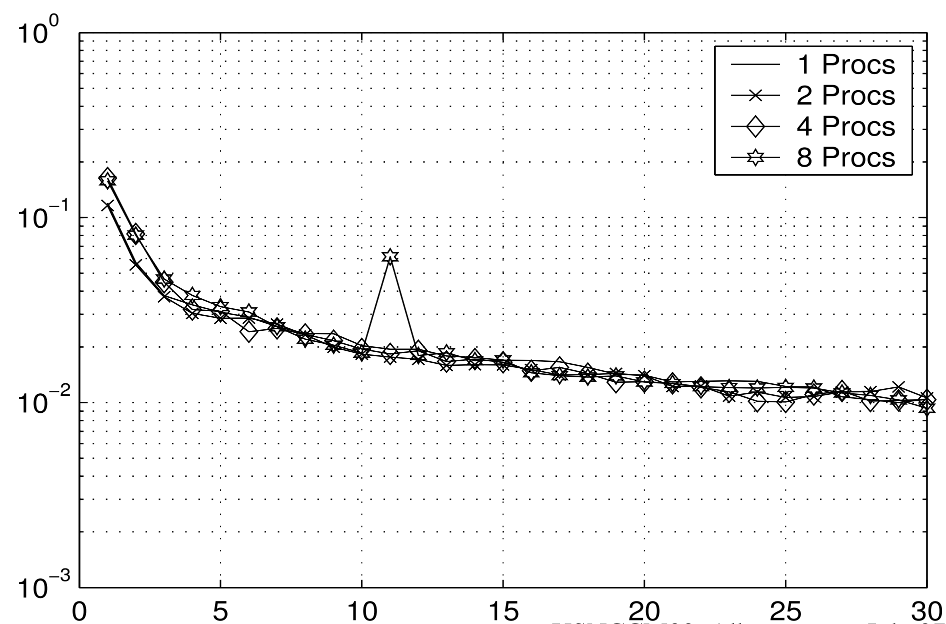
$\#T$	9735	19359	36134	52079	160944
$\varepsilon = \|\|u - u_h\|\|_\infty$	0.025	0.017	0.0095	0.0066	0.0024
$\varepsilon \cdot \#T^{2/3}$	11.3	12.2	10.3	9.1	7.0

Problem 2:

$\#T$	9531	18798	36175	70344	140392
$\varepsilon = \|\|u - u_h\|\|_\infty$	0.057	0.031	0.022	0.016	0.010
$\varepsilon \cdot \#T^{2/3}$	25.5	21.8	23.9	27.1	26.8

Convergence of adaptive iterations:





Numerical experiments (8/8)

Number of mesh modifications and speed-up.

$\#T \sim 160000$.

L	$p=1$	$p=2$		$p=4$		$p=8$	
	#mod	#mod	spd	#mod	spd	#mod	spd
1	30403	16204	1.8	8752	3.4	5484	4.2
2	30850	18273	1.5	12340	2.7	6937	3.7
10	32986	13776	2.9	4996	7.5	1445	12.2

$\#T \sim 140000$.

L	$p=2$	$p=4$	$p=8$
1	1.9	4.2	7.8
10	3.6	9.0	15
20	3.4	10.7	26.6

Conclusions

- There are a few theoretical results for algorithms of generation of quasi-optimal meshes.
- Simple 1D (slice-wise) domain decomposition is acceptable for parallel computers with a small number of processors.
- The most expensive stage of the parallel algorithm is the generation of quasi-optimal subgrids. Communications expenses are negligent.
- The parallel mesh generation may result in super liner speed-up.

Conclusions

- There are a few theoretical results for algorithms of generation of quasi-optimal meshes.
- Simple 1D (slice-wise) domain decomposition is acceptable for parallel computers with a small number of processors.
- The most expensive stage of the parallel algorithm is the generation of quasi-optimal subgrids. Communications expenses are negligent.
- The parallel mesh generation may result in super liner speed-up.

2D and 3D FORTRAN codes are available for research purposes.